Amendments to the Claims

Please amend Claims 1 and 21. The Claim Listing below will replace all prior versions of the claims in the application:

Claim Listing

1. (Currently amended) A data engine, located in a programmable pipeline processor, for processing non-field delineated, streaming, application level database records received from a mass storage device, the data engine comprising:

    a data parser configured to recognize the record and field structure of the non-field delineated database records, determine field boundaries in the non-field delineated database records, and parse the non-field delineated database records into field delineated data, and process fields to select one or more fields to be assembled into output tuples;

    filter logic configured to receive the field delineated data from the data parser, and to filter the field delineated data by performing a field operation on the field delineated data; and

    an output tuple generator configured to assemble filtered field delineated data into an output tuple.

2. (Previously presented) The data engine of claim 1 wherein the filter logic further comprises a programmable memory that serves as a substitution table for field delineated database records, and wherein performing a field operation on the field delineated database records includes performing a field comparison on selected fields of the field delineated data.

3. (Previously presented) The data engine of claim 2 wherein the substitution table includes a data-string register.

4. (Previously presented) The data engine of claim 2 wherein the substitution table includes a temporary register.

5.   (Previously presented) The data engine of claim 2 wherein the field comparison is a character field comparison.

6.-20. (Cancelled)

21.   (Currently amended) A method for processing, in a programmable pipeline processor, non-field delineated, streaming, application level database records received from a mass storage device, the method comprising:

        receiving the non-field delineated database records in a field buffer as an input data stream;

        recognizing the record and field structure of the non-field delineated database records in the field buffer;

        <u>determining field boundaries in the non-field delineated database records by</u> separating the input data stream into field delineated data in the field buffer under instruction from an external central processing unit;

        <u>processing fields to select one or more fields to be assembled into output tuples;</u>

        filtering the field delineated data by sending field delineated data from the field buffer to at least one logic unit that performs at least one field operation on the field delineated data; and

        assembling the filtered field delineated data into an output tuple.

22.   (Previously presented) The data engine of claim 1 wherein the output tuple assembled by the output tuple generator contains only selected data fields of the field delineated data.

23.   (Previously presented) The data engine of claim 1 wherein the filter logic is further configured to filter the field delineated data by flagging data for further processing.

24.   (Previously presented) The data engine of claim 1 further comprising header storage configured to receive header and control data of the field delineated data from the data parser and provide header data to the filter logic, wherein the filter logic is further configured to use header data to filter field delineated data.

25. (Previously presented) The data engine of claim 1 further comprising an ID processing module configured to receive header and control data of the field delineated data, to identify the validity of field delineated data by processing an ID field in the header data, and to provide the identified validity to the tuple generator.

26. (Previously presented) The data engine of claim 25 wherein the ID field stores a transaction ID associated with the field delineated data.

27. (Previously presented) The method of claim 21 wherein filtering further comprises:
    providing a substitution table for field delineated data; and
    performing a field comparison on selected fields of the field delineated data.

28. (Previously presented) The method of claim 27 wherein the substitution table includes a data-string register.

29. (Previously presented) The method of claim 27 wherein the substitution table includes a temporary register.

30. (Previously presented) The method of claim 27 wherein the field comparison is a character field comparison.

31. (Previously presented) The method of claim 21 wherein the output tuple contains only selected data fields of the field delineated data.

32. (Previously presented) The method of claim 21 wherein filtering the field delineated data comprises flagging data for further processing.

33. (Previously presented) The method of claim 21 further comprising using header storage data of the field delineated data to separate field delineated data.

34. (Previously presented) The method of claim 21 further comprising:
    identifying the validity of field delineated data by processing an ID field in header data of the field delineated data; and

assembling the filtered data into the output tuple based on the validity of the field delineated data.

35.   (Previously presented) The method of claim 34 wherein the ID field stores a transaction ID associated with the field delineated data.